Dimension reduction and manifold learning

Multidimensional scaling via classical scaling and stress minimization

Eddie Aamari
Département de mathématiques et applications
CNRS, ENS PSL

Master MASH — Dauphine PSL

Setting

Multidimensional scaling

Multidimensional scaling (MDS) is the term used in psychometry/psychology and statistics to refer to the problem of

Embedding a weighted graph into a Euclidean space.

Graph Embedding

Given a (undirected) weighted graph $(\mathcal{V}, \mathcal{E}, \delta)$ and embedding dimension d, find $y_1, \ldots, y_n \in \mathbb{R}^d$ such that

$$\|y_i - y_j\| \approx \delta_{ij}$$

for all (or most) $(i, j) \in \mathcal{E}$.

$\delta_{ij} =$ dissimilarity between nodes (or *items* / *objects*) i and j.

We will assume that

- $\delta_{ii} = 0$ for all i
- $\delta_{ij} \geq 0$ for all $(i, j) \in \mathcal{E}$
- $\delta_{ij} = \delta_{ji}$ for all $(i, j) \in \mathcal{E}$

Similarities instead ?

Data may consist of *similarities* (or *affinities*/ *proximities*) between nodes.

If this is the case, and a method calls for dissimilarities instead, the usual avenue is to apply a decreasing transformation to the similarities in order to be able to interpret them as dissimilarities.

Remark

The freedom in the choice of transformation makes the situation effectively non-metric, a variant of the embedding problem that we can also be dealt with more advanced methods.

The output of a method takes the form

$$y_1, \ldots, y_n \in \mathbb{R}^d,$$

or equivalently

$$Y := \begin{pmatrix} y_1^\top \\ y_2^\top \\ \vdots \\ y_n^\top \end{pmatrix} \in \mathbb{R}^{n \times d}.$$

The desired embedding dimension d is assumed given except when discussing its choice. (It can be seen as a tuning parameter of any method, although of a special kind.)

Description and Derivation

The main method for MDS is classical scaling (CS).

CS requires that all the dissimilarities be available! (i.e. *complete* graph)

In that case, the input data can be gathered in a dissimilarity matrix

$$\Delta := \begin{pmatrix} \delta_{11} & \delta_{12} & \cdots & \delta_{1n} \\ \delta_{21} & \delta_{22} & \cdots & \delta_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \delta_{n1} & \delta_{n2} & \cdots & \delta_{nn} \end{pmatrix} \in \mathbb{R}^{n \times n}$$

Classical Scaling

Step 1: double-centering the matrix of squared dissimilarities

Form the matrix

$$\Delta_2^c = -\tfrac{1}{2} H \Delta^{\circ 2} H, \quad \text{where } H = I - \tfrac{1}{n} 1 1^\top$$

Step 2: eigendecomposition

Let $\lambda_1 \geq \cdots \geq \lambda_d$ be the top d eigenvalues and $u_1, \ldots, u_d$ be corresponding normed eigenvectors of Δ_2^c

Step 3: embedding

Form the output matrix

$$Y := \left(\sqrt{\max(\lambda_1, 0)}\, u_1 \quad \cdots \quad \sqrt{\max(\lambda_d, 0)}\, u_d \right) \in \mathbb{R}^{n \times d}$$

We first study the method in the realizable case, as this provides a clear motivation.

Definition

$(\mathcal{V}, \mathcal{E}, \delta)$ is realizable in dimension d if there exist $x_1, \ldots, x_n \in \mathbb{R}^d$ such that

$$\delta_{ij} = \|x_i - x_j\| \text{ for all } (i,j) \in \mathcal{E}.$$

Remark

When a graph with n nodes is realizable, it is realizable in dimension $\leq n-1$.

Euclidean point cloud

Let $x_1, \ldots, x_n$ be a (centered w.l.o.g.) point cloud so that

$$\delta_{ij} = \|x_i - x_j\| \text{ and } \bar{x} := \tfrac{1}{n} \textstyle\sum_i x_i = 0.$$

Key idea: Polarize the distances

(i.e. convert dissimilarities (Euclidean distances) into inner products (Gram matrix))

Explaining double centering

We have

$$\delta_{ij}^2 = \|x_i - x_j\|^2 = \langle x_i, x_i\rangle + \langle x_j, x_j\rangle - 2\langle x_i, x_j\rangle.$$

Therefore,

$$\begin{cases} \frac{1}{n}\sum_j \delta_{ij}^2 = \langle x_i, x_i\rangle + \frac{1}{n}\sum_j \langle x_j, x_j\rangle & \text{(sum over } j \text{ with } \bar{x} = 0) \\ \frac{1}{n^2}\sum_i \sum_j \delta_{ij}^2 = \frac{2}{n}\sum_j \langle x_j, x_j\rangle & \text{(sum over } (i,j) \text{ with } \bar{x} = 0) \end{cases}$$

Solving for $\langle x_i, x_j\rangle$ hence yields

$$\begin{aligned} \langle x_i, x_j\rangle &= -\frac{1}{2}(\delta_{ij}^2 - \langle x_i, x_i\rangle - \langle x_j, x_j\rangle) \\ &= -\frac{1}{2}\Big(\delta_{ij}^2 - \frac{1}{n}\sum_l \delta_{il}^2 - \frac{1}{n}\sum_k \delta_{kj}^2 + \frac{1}{n^2}\sum_k\sum_l \delta_{kl}^2\Big) \end{aligned}$$

Writing $X := (x_1|\cdots|x_n)^\top$, the matrix form of the above is $XX^\top = -\frac{1}{2}H\Delta^{\circ 2}H$.

Explaining the eigendecompôsition

This reasoning, *à la* Eckart and Young 1936, yields that the truncated eigenstructure of Δ_2^c provides a best approximation (in any Schatten norm) for a given rank. Therefore, the embedding returned by CS solves:

$$\begin{aligned} \text{minimize} \quad & \|\Delta_2^c - YY^\top\|_F \\ \text{over} \quad & Y \in \mathbb{R}^{n\times d} \end{aligned}$$

Equivalently, if $\Delta_2^c = (b_{ij})_{ij}$,

$$\begin{aligned} \text{minimize} \quad & \text{strain} = \sum_{i,j} (b_{ij} - \langle y_i, y_j\rangle)^2 \\ \text{over} \quad & y_1, \ldots, y_n \in \mathbb{R}^d \end{aligned}$$

Exactness of CS

Definition (Exact method)

A method is exact if it returns a point cloud Y realizing the input dissimilarities.

Note that all the realizing point sets are necessarily rigid transformations of each other.

The derivations above lead to the following (see Torgerson (1952, 1958)).

Theorem (Realizability and rank for CS)

- *Δ is Euclidean* $\quad\Leftrightarrow\quad$ *Δ_2^c is positive semi-definite.*
- *If Δ is Euclidean, it is realizable in dimension*

$$\operatorname{rank}(\Delta_2^c) \leq d \leq n-1,$$

in which case CS is exact.

History of CS

In this form, CS is attributed to Torgerson (1952, 1958), but is based on earlier works

Eckart and Young 1936
(low-rank approximation)
↓
Young and Householder 1938
(via Richardson and Gulliksen)
↓
Torgerson 1952

Rao 1964 and Gower 1966 popularized the method in statistics (connections to PCA).

Other names include *principal coordinates analysis (PCoA)*, *Torgerson Scaling*, and *Torgerson–Gower scaling*.

Perturbation and Consistency

Non-realizable case

CS realizes a graph when possible, and is a way to check when a graph is realizable.

In practice, (e.g. experiments in psychometry), the method has been applied to non-realizable graphs. It can still be applied to output an embedding, but it will not be exact because:

- Δ is Euclidean but the embedding dimension d is (strictly) smaller than $\mathrm{rank}(\Delta_2^c)$; or
- Δ is not Euclidean, and thus not realizable in any dimension.

  If Δ_2^c has negative eigenvalues among the top d, then CS simply discards the corresponding directions, effectively embedding the graph in dimension given by the number of positive eigenvalues.

Handwavy stability

CS is known to practitioners to be stable to noise.

(It is not as stable to outliers. Robust variants available)

This is not surprising for a spectral method

(as eigendecomposition are known to degrade gracefully under perturbation)

For CS, perturbations have been studied in a few papers.

- Sibson 1979 provides some Taylor developments, later refined by de Silva and Tenenbaum 2004.
- Arias-Castro, Javanmard, and Pelletier 2020 obtain true perturbation bounds for classical scaling

Theorem (Stability of CS – Arias-Castro, Javanmard, and Pelletier 2020)

- *Let $X \in \mathbb{R}^{n \times d}$ be centered with radius ρ and half-width ω.*
- *Write $\Xi := (\xi_{ij} := \|x_i - x_j\|)_{ij}$ and $A := 3\sqrt{d}\rho/\omega^2$*
- *Given $\Delta = (\delta_{ij})_{ij}$, set*

$$\eta^4 := \frac{1}{n^2} \sum_{i,j} (\delta_{ij}^2 - \xi_{ij}^2)^2$$

Assume that η is small enough so that $\eta/\omega \leq 1/\sqrt{2}$.

Then CS applied with dimension d to Δ returns a centered point cloud $Y \in \mathbb{R}^{n \times d}$ satisfying

$$\min_{Q \in \mathcal{O}(\mathbb{R}^d)} \left(\frac{1}{n} \sum_{i=1}^{n} \|y_i - Q x_i\|^2 \right)^{1/2} \leq A\eta^2.$$

Comments on perturbative bounds

- Such a perturbation bound only makes sense in a noisy realizable setting.
- But what can be said in a truly non-realizable setting?
  (i.e. where one that cannot be easily compared to a realizable one)
- What about for iid data in a general metric space?
- In such a situation, what does CS really "estimates"?

Probabilistic consistency

Consider:

- A metric space $(\mathbb{Q}, \delta)$ equipped with a (Borel) probability measure P.
- An iid sample $Q_1, \ldots, Q_n \sim_{iid} P$
- Apply CS $\Delta := (\delta(Q_i, Q_j))_{ij}$.

What happens to the output $Y \in \mathbb{R}^{n \times d}$ in the large-sample limit $n \to \infty$?

Questions of consistency have been recently considered by Kroshnin, Stepanov, and Trevisan 2022 and Lim and Memoli 2022, and building on that, by Arias-Castro and Qiao 2022.

Theorem (What CS converges to — Arias-Castro and Qiao 2022)

- $(\mathbb{Q}, \delta)$ *be a compact separable metric space equipped with a (Borel) probability measure* P *, and* $Q_1, \ldots, Q_n, Q, Q' \sim_{iid} P$.
- *Write* $Y = (Y_1 | \cdots | Y_n)^\top \in \mathbb{R}^{n \times d}$ *for the output of* CS *in dimension* d.
- *Set* $b(q, q') := -\frac{1}{2}\left(\delta(q, q')^2 - \mathbb{E}[\delta(q, Q')^2] - \mathbb{E}[\delta(Q, q')^2] + \mathbb{E}[\delta(Q, Q')^2]\right)$

Under mild conditions, there exists $\pi_n : \mathbb{Q} \to \mathbb{R}^d$ *such that* $\pi_n(Q_i) = Y_i$ *a.s, and such that*

$$\mathbb{E}_Q\left[\min_{\pi \in \Pi} \|\pi_n(Q) - \pi(Q)\|^2\right] \xrightarrow[n \to \infty]{} 0,$$

where Π *is the collection of functions minimizing the expected strain, that is*

$$\Pi := \operatorname*{arg\,min}_{\pi : \mathbb{Q} \to \mathbb{R}^d} \mathbb{E}\left[\left(\langle \pi(Q), \pi(Q') \rangle - b(Q, Q')\right)^2\right].$$

Computation and Approximations

Computational complexity of CS

Step 1: double-centering the matrix of squared dissimilarities

- complexity $O(n^2)$

Step 2: eigendecomposition

- complexity $O(n^2 d)$ by a power method

Step 3: embedding

- complexity $O(nd)$

With d small (e.g., $d \leq 3$), the complexity is quadratic in the number of nodes.

Faster variants of CS

Various approximations to CS have been proposed, including

- FastMap by Faloutsos and Lin 1995
- MetricMap by Wang et al. 1999
- Landmark MDS by de Silva and Tenenbaum 2004
  [see also (Kearsley, Tapia, and Trosset 1998; Priyantha et al. 2003)]
- FastMDS by Yang et al. 2006
- split-and-combine MDS by Tzeng, Lu, and Li 2008

The first two are a bit ad hoc. We review the third one. The last two are very similar.

Landmark MDS

Step 1: Select landmark nodes
Uniformly at random among $\binom{n}{k}$ / With farthest point sampling

Step 2: Embed the landmark nodes
This is done by an application of CS.

Step 3: Embed remaining nodes
This is done by a form of lateration (aka external unfolding).
(positioning of a point based on its distances to landmarks)

$\hookrightarrow$ Time complexity $O(kdn)$ for $n \geq k$.

Lateration in vanilla Landmark MDS

- $Y_l \in \mathbb{R}^{k \times d}$ denote a centered point cloud playing the role of the m landmark points.
- Consider a new node with dissimilarities $\delta := (\delta_i) \in \mathbb{R}^{k \times 1}$ to the m landmarks.

Lateration from Landmarks

- Set $\mu_i^2 := \frac{1}{k} \sum_{j=1}^{k} \|x_j - x_i\|^2$, so that $\mu^{\circ 2} \in \mathbb{R}^k$.
- Output position $y := -\frac{1}{2} Y_l^{\dagger} (\delta^{\circ 2} - \mu^{\circ 2})$.

Behind this lateration method: Nyström's formula

Theorem (Nyström / Schur complement)

Let $k < n$, and

$$K = \begin{pmatrix} A & B \\ B^\top & C \end{pmatrix} \in \mathbb{R}^{n \times n}$$

be a symmetric matrix with $A \in \mathbb{R}^{k \times k}$.

- $K \succcurlyeq 0 \iff A \succcurlyeq 0$ *and* $A \succcurlyeq BC^\dagger B^\top$. (*"$a \geq 0$ and $ac - b^2 \geq 0$"*)
- *If* $\operatorname{rank}(A) = \operatorname{rank}(K)$, *then* $C = B^\top A^\dagger B$.

When $\operatorname{rank}(A) < \operatorname{rank}(K)$, the matrix

$$\tilde{K} := \begin{pmatrix} A & B \\ B^\top & B^\top A^\dagger B \end{pmatrix}$$

is called the Nyström approximation of K.

Nyström for lateration with inner products

In the realizable case of Landmark MDS, up to polarization / double centering, we observe inner products $g = (\langle x_i, x \rangle)_{i \leq n}$ of a new point being embedded.

The Gram matrix of $(x_1 | \cdots | x_k | x)^\top \in \mathbb{R}^{(k+1) \times d}$ is given by

$$K := \begin{pmatrix} K_1 & g \\ g^\top & c \end{pmatrix}$$

where $c \in \mathbb{R}$ is unknown.

Assuming that $\mathsf{rank}(K) = \mathsf{rank}(K_1)$, the Nyström's formula gives $c = g^\top K_1^\dagger g$.

($\hookrightarrow$ i.e. we'd like the embedded position of x to be in the span of $Y^\top$)

Nyström for lateration with inner products

Since $K_\mathrm{l} = Y_\mathrm{l} Y_\mathrm{l}^\top$, we have $K_\mathrm{l}^\dagger = (Y_\mathrm{l}^\dagger)^\top Y_\mathrm{l}^\dagger$. Therefore,

$$\tilde{K} := \begin{pmatrix} Y_\mathrm{l} Y_\mathrm{l}^\top & g \\ g^\top & g^\top (Y_\mathrm{l}^\dagger)^\top Y_\mathrm{l}^\dagger g \end{pmatrix} = \tilde{Y} \tilde{Y}^\top,$$

where $\tilde{Y} := (Y_\mathrm{l}^\top | (Y_\mathrm{l}^\dagger g)^\top)^\top \in \mathbb{R}^{(k+1) \times d}$. The embedding of the new point x is thus

$$y_{\text{Lateration}} := Y_\mathrm{l}^\dagger g.$$

Complexity: The evaluation of the new locations only requires a linear amount of computation (in the number of landmarks), and only requires to compute $Y_\mathrm{l}^\dagger$ once.

Nyström for lateration with distances

If only the dissimilarities $\delta = (\|x_i - x\|)_{i \le k} \in \mathbb{R}^{k \times 1}$ are available, we can use the polarization trick. Indeed, since $\sum_{j=1}^{k} x_j = 0$, we have

$$\begin{aligned} 2\langle x, x_i \rangle &= \|x\|^2 + \|x_i\|^2 - \|x - x_i\|^2 \\ &= \frac{1}{k}\sum_j \|x_i - x_j\|^2 + \frac{1}{k}\sum_\ell \|x - x_\ell\|^2 - \frac{1}{k^2}\sum_{k,\ell} \|x_j - x_\ell\|^2 - \|x - x_i\|^2 \\ &= \mu_i^2 + \left(\overline{\delta^{\circ 2}} - \overline{\Delta_{c,\mathrm{l}}^{\circ 2}}\right) - \delta_i^2. \end{aligned}$$

Matricially, $g = -\frac{1}{2}(\delta^{\circ 2} - \mu^{\circ 2} - \left(\overline{\delta^{\circ 2}} - \overline{\Delta_{c,\mathrm{l}}^{\circ 2}}\right)\mathbf{1}_k)$. The Landmark MDS formula becomes

$$y_{\mathsf{LMDS}} = -\frac{1}{2} Y_{\mathrm{l}}^{\dagger}(\delta^{\circ 2} - \mu^{\circ 2} - \left(\overline{\delta^{\circ 2}} - \overline{\Delta_{c,\mathrm{l}}^{\circ 2}}\right)\mathbf{1}_k) = -\frac{1}{2} Y_{\mathrm{l}}^{\dagger}(\delta^{\circ 2} - \mu^{\circ 2}),$$

where we used that $Y_{\mathrm{l}}^{\dagger}\mathbf{1}_k = 0$.

Self-consistency of Landmark MDS

Landmark MDS. yields a lateration method that is exact when it is possible to be exact.

Theorem

If $Y_1 \in \mathbb{R}^{k \times d}$ is full rank and centered, then the Landmark MDS algorithm is consistent with the embedded landmarks.

That is, if $\delta = (\|y_i - y_j\|)_j$ for some $i \in \{1, \ldots, k\}$, then the algorithm returns y_i.

Arias-Castro, Javanmard, and Pelletier (2020) obtain a perturbation bound for this form of lateration, which is then used to derive a perturbation bound for Landmark MDS.

Step 1: Partition the node set
Partition the set of nodes into subsets of approximately same size, say, k subsets of size about m (so that $k \times m \approx n$).

Step 2: Embed each subset
Embed each subset by CS obtaining 'patches'.

Step 3: Align the patches
This is done by selecting s landmark nodes from each subset, embedding all these $k \times s$ landmark nodes together by CS. Then align each patch with the corresponding landmarks by procrustes.

To Python!

Stress

Element-wise formulation of CS

As noted above, the embedding returned by CS solves the least squares problem

$$\begin{aligned} \text{minimize} \quad & \|\Delta_2^c - YY^\top\|_F^2 \\ \text{over} \quad & Y \in \mathbb{R}^{n \times d} \end{aligned}$$

or, equivalently, if $\Delta_2^c = (b_{ij})$,

$$\begin{aligned} \text{minimize} \quad & \text{strain} = \sum_{i<j} (b_{ij} - \langle y_i, y_j \rangle)^2 \\ \text{over} \quad & y_1, \ldots, y_n \in \mathbb{R}^d \end{aligned}$$

Find $y_1, \ldots, y_n \in \mathbb{R}^d$ such that

$$\|y_i - y_j\| \approx \delta_{ij}$$

for all (or most) $(i, j) \in \mathcal{E}$.

A more direct formulation as an optimization problem would lead, for example, to solving (Kruskal 1964a,b)

$$\begin{aligned} \text{minimize} \quad & \text{stress} = \sum_{(i,j)\in\mathcal{E}} (\delta_{ij} - \|y_i - y_j\|)^2 \\ \text{over} \quad & y_1, \ldots, y_n \in \mathbb{R}^d \end{aligned}$$

Kruskal's stress

Other variants include smoother functionals such as (Takane, Young, and De Leeuw 1977)

$$\text{s-stress} = \sum_{(i,j)\in\mathcal{E}} \left(\delta_{ij}^2 - \|y_i - y_j\|^2\right)^2$$

and more robust functionals such as (Heiser 1988)

$$\text{absolute stress} = \sum_{(i,j)\in\mathcal{E}} \left|\delta_{ij} - \|y_i - y_j\|\right|$$

Fifty shades of stress I

Some notions of stress are based on ratios instead, such as (Ramsay 1982)

$$\text{multiscale stress} = \sum_{(i,j)\in\mathcal{E}} \log^2(\|y_i - y_j\|/\delta_{ij})$$

which to first order coincides with (McGee 1966)

$$\begin{aligned}\text{elastic stress} &= \sum_{(i,j)\in\mathcal{E}} \big(1 - \|y_i - y_j\|/\delta_{ij}\big)^2 \\ &= \sum_{(i,j)\in\mathcal{E}} w_{ij}\big(\delta_{ij} - \|y_i - y_j\|\big)^2, \quad w_{ij} := 1/\delta_{ij}^2\end{aligned}$$

Fifty shades of stress II

Other notions of weighted stress include (Sammon 1969)

$$\text{Sammon stress} = \sum_{(i,j)\in\mathcal{E}} w_{ij}\big(\delta_{ij} - \|y_i - y_j\|\big)^2, \quad w_{ij} := 1/\delta_{ij}$$

Renormalized stress

These are in fact notions of *raw stress*, as they are sometimes normalized. For example, Kruskal 1964a,b defines the *normalized stress* as

$$\sqrt{\frac{\sum_{(i,j)\in\mathcal{E}}(\delta_{ij} - \|y_i - y_j\|)^2}{\sum_{(i,j)\in\mathcal{E}} \delta_{ij}^2}}$$

This version can provide a measure of quality of fit.

Optimization

Take, for example, Kruskal's stress

$$\sigma(y_1, \dots, y_n) := \sum_{(i,j)\in\mathcal{E}} (\delta_{ij} - \|y_i - y_j\|)^2$$

Our goal is to solve

$$\begin{aligned} \text{minimize} \quad & \sigma(y_1, \dots, y_n) \\ \text{over} \quad & y_1, \dots, y_n \in \mathbb{R}^d \end{aligned}$$

Everything that follows extends in a straightforward manner to any weighted stress

$$\sum_{(i,j)\in\mathcal{E}} w_{ij}(\delta_{ij} - \|y_i - y_j\|)^2.$$

Optimization

The problem is:

- non-convex (⚠ Composition of convex functions possibly not convex ⚠)
- high-dimensional: $S : \mathbb{R}^{n \times d} \to \mathbb{R}$

Even for $d = 1$ where it is sometimes called unidimensional scaling, this minimization problem is known to be NP-hard (De Leeuw and Heiser 1977).

(Closely related to the problem of seriation, which is known to be NP-hard.)

Gradient descent

Gradient descent

Kruskal 1964b proposes a first order gradient descent procedure.

$$y_i^0 \leftarrow \text{initialization}$$
$$y_i^{t+1} \leftarrow y_i^t - \rho_t \nabla_{y_i} \sigma(y_1^t, \ldots, y_n^t)$$

Simple calculations give

$$\nabla_{y_i} \sigma(y_1, \ldots, y_n) = 2 \sum_{j \sim i} \frac{y_i - y_j}{\|y_i - y_j\|} (\|y_i - y_j\| - \delta_{ij})$$

and the step size ρ_t may depend on $(y_1^t, \ldots, y_n^t)$, in particular through the stress, and is generally made to decrease.

Surprisingly, he seems to recommend a *random initialization*, rather than using the output of classical scaling. (already established as the main method for MDS at the time)

Gradient descent matricially

Recall that $Y = (y_1|\cdots|y_n)^\top$ and work with $\sigma(Y)$, to derive

$$\tfrac{1}{2}\nabla\sigma(Y) = VY - B(Y)Y$$

where

$$V := \sum_{(i,j)\in\mathcal{E}} A_{ij} \qquad\qquad B(Y) := \sum_{(i,j)\in\mathcal{E}} \frac{\delta_{ij}}{d_{ij}(Y)} A_{ij}$$

with

$$A_{ij} := (e_i - e_j)(e_i - e_j)^\top \qquad\qquad d_{ij}(Y) := \|y_i - y_j\|$$

$e_1, \ldots, e_n$ being the canonical basis for $\mathbb{R}^n$.

Fixed point method

Guttman 1968 looks at the equation giving the stationary points:

$$\nabla\sigma(Y) = 0 \;\Leftrightarrow\; VY = B(Y)Y \;\Leftrightarrow\; Y = V^{\dagger}B(Y)Y.$$

He hence proposes use the iterated fixed-point method

$$Y^{t+1} \leftarrow V^{\dagger}B(Y^t)Y^t.$$

It turns out that this is a form of gradient descent:

$$Y^{t+1} \leftarrow Y^t - \tfrac{1}{2}V^{\dagger}\nabla\sigma(Y^t).$$

De Leeuw 1988 establishes convergence of this procedure to stationary points.

Second order methods

Kamada and Kawai 1989 apply the Newton–Raphson (2nd order) procedure to minimizing the stress.

Other Newton and quasi-Newton variant procedures are applied, e.g., in (Glunt, Hayden, and Raydan 1993; Kearsley, Tapia, and Trosset 1998).

Augmentation, Majorization, Alternate Minimization

Augmentation

Augmentation consists in introducing new variables to make some iterations more straightforward. (Think EM algorithm.)

In general, suppose we want to solve

$$\min_y f(y)$$

The general idea is to find a auxiliary function g satisfying

$$f(y) = \min_z g(y, z)$$

such that both $\min_y g(y, z)$ and $\min_z g(y, z)$ are is relatively easy to compute.

Alternate minimization

Note that

$$\min_y f(y) = \min_y \min_z g(y,z) = \min_z \min_y g(y,z)$$

This inspires an alternate minimization approach:

$$\begin{aligned} y^0 &\leftarrow \text{initialization} \\ z^t &\leftarrow \arg\min_z g(y^{t-1}, z) \\ y^t &\leftarrow \arg\min_y g(y, z^t) \end{aligned}$$

In this scheme, the successive values of g are monotone:

$$g(y^t, z^t) = \min_y g(y, z^t) \leq g(y^{t-1}, z^t) = \min_z g(y^{t-1}, z) \leq g(y^{t-1}, z^{t-1})$$

Leeuw 1975 proposes an augmentation approach called Elegant.

See (Browne 1987) for a related alternate minimization approach.

Working with the *s-stress*, define

$$\sigma(Y, \delta) := \sum_{i,j,k,l} \big(\delta_{ijkl} - d_{ijkl}(Y)\big)^2$$
$$\text{with } d_{ijkl}(Y) := (y_i - y_j)^\top (y_k - y_l)$$

where $\delta_{ijij} = \delta_{ij}^2$ is enforced throughout and Y remains centered.

(As it turns out, there is no need to store a 4-way tensor.)

Minimization over the δ_{ijkl} (with Y fixed) is straightforward:

$$\delta_{ijkl} = \begin{cases} \delta_{ij}^2 & \text{if } (i,j) = (k,l) \\ d_{ijkl} & \text{otherwise} \end{cases}$$

For the minimization over the Y (with the δ_{ijkl} fixed)

$$\begin{aligned}\sigma(Y,\delta) &= \sum_{i,j,k,l}\delta_{ijkl}^2 - 2\sum_{i,j,k,l}\delta_{ijkl}d_{ijkl}(Y) + \sum_{i,j,k,l}d_{ijkl}(Y)^2 \\ &= \mathsf{fun}(\delta) - 8n^2\sum_{i,j,k,l}\operatorname{trace}(UYY^\top) + 4n^2\operatorname{trace}((YY^\top)^2) \\ &\qquad \text{where } u_{ij} := \tfrac{1}{4n^2}\sum_{k,l}(\delta_{ikjl} - \delta_{ilkj} - \delta_{kijl} + \delta_{likj}) \\ &= \mathsf{fun}(\delta) + 4n^2\operatorname{trace}((YY^\top - U)^2)\end{aligned}$$

Y is obtained by a truncated eigendecomposition of U as in CS but with modified dissimilarities that change with each iteration.

Majorization

Majorization — aka *Majorization-Minimization* (Mairal 2015) — is a special form of augmentation where

$$f(y) = g(y, y)$$

meaning that the minimization

$$f(y) = \min_z g(y, z)$$

is attained at y itself.

(Note that the y and z variables belong here to the 'same' space.)

Sandwich inequality

The scheme simplifies to

$$y^0 \leftarrow \text{initialization}$$
$$y^t \leftarrow \arg\min_y g(y, y^{t-1})$$

And the successive values of f are monotone:

$$f(y^t) = \min_y g(y^t, y) \leq g(y^t, y^{t-1}) := \min_y g(y, y^{t-1}) \leq g(y^{t-1}, y^{t-1}) = f(y^{t-1})$$

Sequence of inequalities called sandwich inequality by De Leeuw 1977.

Scaling by MAjorizing a COmplicated Function (SMACOF)

De Leeuw 1977 proposes a majorization approach to minimizing the stress today known as SMACOF (De Leeuw and Mair 2009).

Recalling that $d_{ij}(Y) = \|y_i - y_j\|$, we have

$$\begin{aligned}\sigma(Y) &= \sum_{(i,j)\in\mathcal{E}} (\delta_{ij} - d_{ij}(Y))^2 \\ &= \sum_{(i,j)\in\mathcal{E}} \delta_{ij}^2 - 2\sum_{(i,j)\in\mathcal{E}} \delta_{ij} d_{ij}(Y) + \sum_{(i,j)\in\mathcal{E}} d_{ij}(Y)^2\end{aligned}$$

The key idea is the following application of the Cauchy–Schwarz inequality. For all $Z = (z_1 \cdots z_n)^\top \in \mathbb{R}^{n\times d}$,

$$\langle y_i - y_j, z_i - z_j\rangle \leq d_{ij}(Y) d_{ij}(Z).$$

Plugging this inequality in the second sum, we have

$$\begin{aligned}\sigma(Y) &\leq \sum_{(i,j)\in\mathcal{E}} \delta_{ij}^2 - 2\sum_{(i,j)\in\mathcal{E}} \zeta_{ij}(Z)^\top (y_i - y_j) + \sum_{(i,j)\in\mathcal{E}} \|y_i - y_j\|^2 \\ &=: \tau(Y, Z)\end{aligned}$$

where $\zeta_{ij}(Z) := \delta_{ij}(z_i - z_j)/\|z_i - z_j\|$.

And, by construction,

$$\sigma(Y) = \tau(Y, Y)$$

Hence, τ is majorizing, and the corresponding scheme is

$$
\begin{aligned}
Y^0 &\leftarrow \text{initialization} \\
Y^t &\leftarrow \arg\min_Y \tau(Y, Y^{t-1})
\end{aligned}
$$

which is attractive since $Y \mapsto \tau(Y, Z)$ is (convex) quadratic.

As it turns out, the scheme coincides with Guttman 1968's, because (with the same notation as before)

$$\nabla_Y \tau(Y, Z) = 0 \;\Leftrightarrow\; VY = B(Z)Z \;\Leftrightarrow\; Y = V^\dagger B(Z)Z$$

Majorizing the absolute stress

Heiser 1988 designs a majorization scheme for the absolute stress:

$$\begin{aligned}\sigma(Y) &:= \sum_{(i,j)\in\mathcal{E}} \left|\delta_{ij} - d_{ij}(Y)\right| \\ &= \sum_{(i,j)\in\mathcal{E}} a_{ij}(Y), \quad a_{ij}(Y) := \left|\delta_{ij} - d_{ij}(Y)\right|\end{aligned}$$

The basic inequality is, for any other configuration $Z \in \mathbb{R}^{n\times d}$,

$$2a_{ij}(Y) \leq a_{ij}(Z) + a_{ij}(Y)^2/a_{ij}(Z)$$

which gives

$$2\sigma(Y) \leq \sigma(Z) + \sum_{(i,j)\in\mathcal{E}} b_{ij}(Z)a_{ij}(Y)^2, \quad b_{ij}(Z) := 1/a_{ij}(Z)$$

When Z is fixed, only the second term on the RHS matters and, being a weighted stress, can be majorized as done before.

Coordinate descent

Coordinate descent is also a form of alternate minimization. In its most basic form, it tackles a multivariate minimization problem such as

$$\min_y f(y) \equiv \min_{y_1,\dots,y_p} f(y_1,\dots,y_p)$$

simply by proceeding one variable (or sometimes a batch of variables) at a time, justified by

$$\min_{y_1,\dots,y_p} f(y_1,\dots,y_p) \equiv \min_{y_{\pi_1}} \cdots \min_{y_{\pi_p}} f(y_1,\dots,y_p)$$

for any permutation $(\pi_1,\dots,\pi_p)$ of $(1,\dots,p)$.

(In our case, $p = nd$)

Position descent

In stress minimization, such a procedure often operates at the level of positions, with each position corresponding to d real variables when embedding in dimension d — a variant which could be called position descent.

- This is implicitly what Shepard 1962 does, and explicitly what Agrafiotis 2003 does (in random order) to minimize the stress.
- Kamada and Kawai 1989 employ a variant of position descent in their Newton–Raphson implementation.
- Gansner, Koren, and North 2004, Costa, Patwari, and Hero III 2006, Zhang et al. 2010 propose position descent variants of majorization.

About initialization

All these iterative approaches require some initialization, i.e. a starting configuration. Two main ways:

- *Random initialization* Draw n points at random from a distribution supported on $\mathbb{R}^d$ (e.g., uniform in $[0,1]^d$).
- *Classical scaling* This option is available if all dissimilarities are provided (i.e., if the graph is complete).

Some very limited numerical experiments in the noisy realizable setting indicate that the result is very similar, although with a random initialization $10\times$ more iterations are needed for convergence.

Iterative approaches may be seen as providing a refinement to an embedding produced by a method like CS or any of the other methods that follow.

Incremental approaches

Incremental constructions

A variety of incremental approaches have been proposed, for example, by Bronstein et al. 2006; Cohen 1997; Williams and Munzner 2004 — among others (Klimenta 2012, Sec 7.1.1).

The general approach of Cohen 1997 and Williams and Munzner 2004 operates in batches by:

1. Positioning the first batch, say, by applying CS;
2. For a subsequent batch, initialize each node at the same location as its closest neighbor in the graph, and then applying an iterative method, say, SMACOF.

Bronstein et al. 2006 develop a multigrid scheme.

Beyond Landmark MDS

Another, distinct line of incremental methods implement flavors of sequential lateration.

Recall Landmark MDS (de Silva and Tenenbaum 2004):

1. Select 'landmark' nodes (at random);
2. Embed the landmark nodes (by CS);
3. Embed remaining nodes (by Gower 1968's lateration method).

What if not all nodes are reached in that way?

In sequential lateration (Aspnes et al. 2006; Bakonyi and Johnson 1995; Eren et al. 2004; Fang et al. 2009; Grone et al. 1984; Kearsley, Tapia, and Trosset 1998; Laurent 2001) the process is simply iterated, with the embedded points playing the role of landmark points in the next iteration.

Sequential Lateration (Aspnes et al. 2006)

- *Step 0:* Select a *complete subgraph* $\mathcal{V}^0$ with at least $d+1$ nodes (at random) and embed it using CS
- *Step t:* Laterize any node with at least $d+1$ neighbors in $\mathcal{V}^{t-1}$ and add those nodes to $\mathcal{V}^{t-1}$ to form $\mathcal{V}^t$

Lateration graphs

Sequential lateration can operate even when some dissimilarities are missing (i.e., the graph is not complete). When it is exact, the graph is a lateration graph (Aspnes et al. 2006).

Under mild conditions, a large random geometric graph is a lateration graph with high probability (Arias-Castro and Chau 2022; Aspnes et al. 2006).

In recent work (Arias-Castro and Chau 2022), we derive a perturbation bound (with an implicit constant) for sequential lateration in a noisy realizable setting built on a lateration graph.

Divide-and-Conquer

Patch-based methods

The incremental approaches already have a *divide-and-conquer* flavor to them, but here we use that qualification for methods based on patches (embedded subgraphs).

We saw two such methods:

- FastMDS (Yang et al. 2006)
- split-and-combine MDS (Tzeng, Lu, and Li 2008)

However, these assume that all dissimilarities are available: Their main motivation is computational.

Divide-and-Conquer

We are interested in such approaches to better deal with situation where there are missing dissimilarities.

DIVIDE-AND-CONQUER (prototypical)

- Select a covering of the graph by *complete subgraphs*, each with at least $d + 1$ nodes, and embed them (by CS)
- Align the patches (by a form of generalized Procrustes)

Variants of this general approach are proposed in (Cucuringu, Lipman, and Singer 2012; Drusvyatskiy et al. 2017; Hendrickson 1995; Koren, Gotsman, and Ben-Chen 2005; Krislock and Wolkowicz 2010; Shang and Ruml 2004; Singer 2008; Zhang et al. 2010), among other works.

References

Agrafiotis, Dimitris K (2003). **"Stochastic proximity embedding".** In: *Journal of computational chemistry* 24.10, pp. 1215–1221.

Arias-Castro, Ery and Phong Alain Chau (2022). **"Supervising embedding algorithms using the stress".** In: *Arxiv preprint arxiv:2207.07218*.

Arias-Castro, Ery, Adel Javanmard, and Bruno Pelletier (2020). **"Perturbation bounds for procrustes, classical scaling, and trilateration, with applications to manifold learning".** In: *Journal of machine learning research* 21, pp. 1–37.

Arias-Castro, Ery and Wanli Qiao (2022). **"Embedding functional data: multidimensional scaling and manifold learning".** In: *Arxiv preprint arxiv:2208.14540*.

Aspnes, James, Tolga Eren, David Kiyoshi Goldenberg, A Stephen Morse, Walter Whiteley, Yang Richard Yang, Brian DO Anderson, and Peter N Belhumeur (2006). **"A theory of network localization".** In: *Ieee transactions on mobile computing* 5.12, pp. 1663–1678.

Bakonyi, Mihály and Charles R Johnson (1995). **"The euclidian distance matrix completion problem".** In: *Siam journal on matrix analysis and applications* 16.2, pp. 646–654.

Bronstein, Michael M, Alexander M Bronstein, Ron Kimmel, and Irad Yavneh (2006). **"Multigrid multidimensional scaling".** In: *Numerical linear algebra with applications* 13.2-3, pp. 149–171.

Browne, MW (1987). **"The young-householder algorithm and the least squares multidimensional scaling of squared distances".** In: *Journal of classification* 4.2, pp. 175–190.

Cohen, Jonathan D (1997). **"Drawing graphs to convey proximity: an incremental arrangement method".** In: *Acm transactions on computer-human interaction (tochi)* 4.3, pp. 197–229.

Costa, Jose A, Neal Patwari, and Alfred O Hero III (2006). **"Distributed weighted-multidimensional scaling for node localization in sensor networks".** In: *Acm transactions on sensor networks (tosn)* 2.1, pp. 39–64.

Cucuringu, Mihai, Yaron Lipman, and Amit Singer (2012). **"Sensor network localization by eigenvector synchronization over the euclidean group".** In: *Acm transactions on sensor networks (tosn)* 8.3, p. 19.

De Leeuw, Jan (1977). **"Applications of convex analysis to multidimensional scaling".** In: *Recent developments in statistics*. Ed. by J.R. Barra, F. Brodeau, G. Romier, and B. van Cutsem. North-Holland Publishing Company.

— (1988). **"Convergence of the majorization method for multidimensional scaling".** In: *Journal of classification* 5.2, pp. 163–180.

De Leeuw, Jan and Willem J Heiser (1977). **"Convergence of correction matrix algorithms for multidimensional scaling".** In: *Geometric representations of relational data*. Ed. by J. C. Lingoes. Vol. 36. Mathesis Press, pp. 735–752.

De Leeuw, Jan and Patrick Mair (2009). **"Multidimensional scaling using majorization: smacof in r".** In: *Journal of statistical software* 31.i03.

de Silva, Vin and Joshua B Tenenbaum (2004). **Sparse multidimensional scaling using landmark points.** Tech. rep. Technical report, Stanford University.

Drusvyatskiy, Dmitriy, Nathan Krislock, Y-L Voronin, and Henry Wolkowicz (2017). **"Noisy Euclidean distance realization: robust facial reduction and the Pareto frontier".** In: *Siam journal on optimization* 27.4, pp. 2301–2331.

Eckart, Carl and Gale Young (1936). **"The approximation of one matrix by another of lower rank".** In: *Psychometrika* 1.3, pp. 211–218.

Eren, Tolga, OK Goldenberg, Walter Whiteley, Yang Richard Yang, A Stephen Morse, Brian DO Anderson, and Peter N Belhumeur (2004). **"Rigidity, computation, and randomization in network localization".** In: *Infocom 2004. twenty-third annualjoint conference of the ieee computer and communications societies*. Vol. 4. IEEE, pp. 2673–2684.

Faloutsos, Christos and King-Ip Lin (1995). **"Fastmap: a fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets".** In: *Proceedings of the 1995 acm sigmod international conference on management of data*, pp. 163–174.

Fang, Jia, Ming Cao, A Stephen Morse, and Brian DO Anderson (2009). **"Sequential localization of sensor networks".** In: *Siam journal on control and optimization* 48.1, pp. 321–350.

Gansner, Emden R, Yehuda Koren, and Stephen North (2004). **"Graph drawing by stress majorization".** In: *International symposium on graph drawing*. Springer, pp. 239–250.

Glunt, W, Tom L Hayden, and Marcos Raydan (1993). **"Molecular conformations from distance matrices".** In: *Journal of computational chemistry* 14.1, pp. 114–120.

Gower, John C (1966). **"Some distance properties of latent root and vector methods used in multivariate analysis".** In: *Biometrika* 53.3-4, pp. 325–338.

— (1968). **"Adding a point to vector diagrams in multivariate analysis".** In: *Biometrika* 55.3, pp. 582–585.

Grone, Robert, Charles R Johnson, Eduardo M Sá, and Henry Wolkowicz (1984). **"Positive definite completions of partial hermitian matrices".** In: *Linear algebra and its applications* 58, pp. 109–124.

Guttman, Louis (1968). **"A general nonmetric technique for finding the smallest coordinate space for a configuration of points".** In: *Psychometrika* 33.4, pp. 469–506.

Heiser, William J (1988). **"Multidimensional scaling with least absolute residuals".** In: *Classification and related methods of data analysis*, pp. 455–462.

Hendrickson, Bruce (1995). **"The molecule problem: exploiting structure in global optimization".** In: *Siam journal on optimization* 5.4, pp. 835–857.

Kamada, Tomihisa and Satoru Kawai (1989). **"An algorithm for drawing general undirected graphs".** In: *Information processing letters* 31.1, pp. 7–15.

Kearsley, Anthony J, Richard A Tapia, and Michael W Trosset (1998). **"The solution of the metric STRESS and SSTRESS problems in multidimensional scaling using Newton's method".** In: *Computational statistics* 13.3, pp. 369–396.

Klimenta, Mirza (2012). **"Extending the usability of multidimensional scaling for graph drawing".** PhD thesis. Universität Konstanz.

Koren, Yehuda, Craig Gotsman, and Mirela Ben-Chen (2005). **PATCHWORK: Efficient localization for sensor networks by distributed global optimization.** Tech. rep.

Krislock, Nathan and Henry Wolkowicz (2010). **"Explicit sensor network localization using semidefinite representations and facial reductions".** In: *Siam journal on optimization* 20.5, pp. 2679–2708.

Kroshnin, Alexey, Eugene Stepanov, and Dario Trevisan (2022). **"Infinite multidimensional scaling for metric measure spaces".** In: *Arxiv preprint arxiv:2201.05885*.

Kruskal, J. B. (1964a). **"Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis".** In: *Psychometrika* 29, pp. 1–27.

— (1964b). **"Nonmetric multidimensional scaling: a numerical method".** In: *Psychometrika* 29.2, pp. 115–129.

Laurent, Monique (2001). **"Polynomial instances of the positive semidefinite and euclidean distance matrix completion problems".** In: *Siam journal on matrix analysis and applications* 22.3, pp. 874–894.

Leeuw, Jan de (1975). **An alternating least squares approach to squared distance scaling.** Tech. rep. Department of Data Theory FSW/RUL.

Lim, Sunhyuk and Facundo Memoli (2022). **"Classical MDS on metric measure spaces".** In: *Arxiv preprint arxiv:2201.09385*.

Mairal, Julien (2015). **"Incremental majorization-minimization optimization with application to large-scale machine learning".** In: *Siam journal on optimization* 25.2, pp. 829–855.

McGee, Victor E (1966). **"The multidimensional analysis of 'elastic' distances".** In: *British journal of mathematical and statistical psychology* 19.2, pp. 181–196.

Priyantha, Nissanka B, Hari Balakrishnan, Erik Demaine, and Seth Teller (2003). **"Anchor-free distributed localization in sensor networks".** In: *Proceedings of the 1st international conference on embedded networked sensor systems*, pp. 340–341.

Ramsay, James O (1982). **"Some statistical approaches to multidimensional scaling data".** In: *Journal of the royal statistical society: series a (general)* 145.3, pp. 285–303.

Rao, C Radhakrishna (1964). **"The use and interpretation of principal component analysis in applied research".** In: *Sankhyā: the indian journal of statistics, series a*, pp. 329–358.

Sammon, John W (1969). **"A nonlinear mapping for data structure analysis".** In: *Ieee transactions on computers* 100.5, pp. 401–409.

Shang, Yi and Wheeler Ruml (2004). **"Improved mds-based localization".** In: *Conference of the ieee computer and communications societies*. Vol. 4. IEEE, pp. 2640–2651.

Shepard, Roger N. (1962). **"The analysis of proximities: multidimensional scaling with an unknown distance function. I".** In: *Psychometrika* 27, pp. 125–140.

Sibson, Robin (1979). **"Studies in the robustness of multidimensional scaling: perturbational analysis of classical scaling".** In: *Journal of the royal statistical society. series b (methodological)*, pp. 217–229.

Singer, Amit (2008). **"A remark on global positioning from local distances".** In: *Proceedings of the national academy of sciences* 105.28, pp. 9507–9511.

Takane, Yoshio, Forrest W Young, and Jan De Leeuw (1977). **"Nonmetric individual differences multidimensional scaling: an alternating least squares method with optimal scaling features".** In: *Psychometrika* 42.1, pp. 7–67.

Torgerson, Warren S (1952). **"Multidimensional scaling: i. theory and method".** In: *Psychometrika* 17.4, pp. 401–419.

— (1958). **Theory and methods of scaling.** Wiley.

Tzeng, Jengnan, Henry Horng-Shing Lu, and Wen-Hsiung Li (2008). **"Multidimensional scaling for large genomic data sets".** In: *Bmc bioinformatics* 9.1, pp. 1–17.

Wang, Jason Tsong-Li, Xiong Wang, King-Ip Lin, Dennis Shasha, Bruce A Shapiro, and Kaizhong Zhang (1999). **"Evaluating a class of distance-mapping algorithms for data mining and clustering".** In: *Proceedings of the fifth acm sigkdd international conference on knowledge discovery and data mining*, pp. 307–311.

Williams, Matt and Tamara Munzner (2004). **"Steerable, progressive multidimensional scaling".** In: *Ieee symposium on information visualization*. IEEE, pp. 57–64.

Yang, Tynia, Jinze Liu, Leonard McMillan, and Wei Wang (2006). **"A fast approximation to multidimensional scaling".** In: *Ieee workshop on computation intensive methods for computer vision*.

Young, Gale and Aiston S Householder (1938). **"Discussion of a set of points in terms of their mutual distances".** In: *Psychometrika* 3.1, pp. 19–22.

Zhang, Lei, Ligang Liu, Craig Gotsman, and Steven J Gortler (2010). **"An as-rigid-as-possible approach to sensor network localization".** In: *Acm transactions on sensor networks (tosn)* 6.4, p. 35.